REMARKS

The Office Action dated February 20, 2008 has been received and carefully noted. The above amendments to the claims, and the following remarks, are submitted as a full and complete response thereto.

Claims 1, 6 and 10 have been amended to more particularly point out and distinctly claim the invention. No new matter has been added. Claims 1-13 are submitted for consideration.

Claims 1-4, 6-8 and 10-12 were rejected under 35 U.S.C. 103(a) as being obvious over European Patent No. 0 572 145 to Thompson (hereinafter Thompson) in view of U.S. Patent No. 6,512,773 to Scott (hereinafter Scott) and further in view of U.S. Patent No. 7,139,271 to Parruck (hereinafter Parruck). According to the Office Action, Thompson teaches all of the elements of claims 1-4, 6-8 and 10-12 except for a data packet including a plurality of cells including a header cell, wherein the header cell of the plurality of cells includes a header and a packet data portion and a counter to determine the number of bytes of a packet after the header has been removed. Thus, the Office Action uses Scott and Parruck to cure the deficiencies of Thompson in an effort to yield the combination of elements recited in claims 1-4, 6-8 and 10-12. The rejection is traversed as being based on references that neither teach nor suggest the novel combination of features clearly recited in claims 1-4, 6-8 and 10-12.

Claim 1, upon which claims 2-5 depend, recites a network device that is configured to prevent data misalignment of a data packet containing extra header bytes.

The network device includes an ingress module having an input interface to receive a data packet including a plurality of cells. A header cell of the data packet is one of the plurality of cells of the data packet. The header cell of the plurality of cells includes a header and packet data information. The header cell includes the header in its entirety for the data packet. The device also includes a header detector configured to detect the header cell of the data packet and remove the header from the header cell of the data packet. The network device also includes a counter configured to determine whether the header cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed from the header cell. The network device further includes an insertion module configured to insert null bytes into the header cell of the data packet to form a modified header cell of the data packet if the counter determines that the header cell of the data packet does not satisfy the multiple of the predetermined number of bytes in order to align all of the other cells of the packet. The network device also includes an extraction module configured to remove the null bytes from the modified header cell of the data packet as a modified cell of the data packet exits the network device.

Claim 6, upon which claims 7-9 depend, recites a method of preventing data misalignment of a data packet containing extra header bytes. The method includes receiving, at an input port of a network device, a data packet including a plurality of cells. A header cell of the data packet is one of the plurality of cells of the data packet. The header cell of the plurality of cells includes a header and packet data information. The

header cell includes the header in its entirety for the data packet. The method also includes detecting the header cell of the data packet, removing the header from the header cell of the data packet and determining whether the header cell of the data packet contains a multiple of a predetermined number of bytes after the header has been removed from the header cell. The method further includes inserting null bytes into the header cell of the data packet to form a modified header cell of the data packet if the counter determines that the header cell of the data packet does not satisfy the multiple of the predetermined number of bytes in order to align all of the other cells of the packet and forwarding the modified cell of the data packet to an output port. The method also includes removing the null bytes from the modified header cell of the data packet as a modified cell of the data packet exits the network device.

Claim 10, upon which claims 11-13 depend, recites a network device configured to prevent data misalignment of a data packet containing extra header bytes. The network device includes receiving means for receiving, at an input port of a network device, a data packet including a plurality of cells. A header cell of the data packet is one of the plurality of cells of the data packet. The header cell of the plurality of cells includes a header and packet data information. The header cell includes the header in its entirety for the data packet. The network device also includes detecting means for detecting the header cell of the data packet. The network device also includes header removing means for removing the header from the header cell of the data packet and determining means for determining whether the header cell of the data packet contains a

multiple of a predetermined number of bytes after the header has been removed from the header cell. The network device further includes inserting means for inserting null bytes into the header cell packet to form a modified header cell of the data packet if the counter determines that the cell of the data packet does not satisfy the multiple of the predetermined number of bytes in order to align all of the other cells of the packet and forwarding means for forwarding the modified cell of the data packet to an output port. The network device also include null byte removing means for removing the null bytes from the modified header cell of the data packet as a modified cell of the data packet exits the network device.

As will be discussed below, the cited prior art references of Thompson, Scott and Parruck fail to disclose or suggest the elements of any of the presently pending claims.

Thompson teaches a computer system with a processor, a cache, a memory and a network adapter. The network adapter generates and inserts network data checksums. In the outbound direction, the processor provides checksum control information to the network adapter and the network adapter calculates the checksum and inserts the checksum into the proper location within the packet before transmitting the packet on the network. In the inbound direction, the network adapter decodes the packet header, programs the checksum control information directly into internal registers, calculates the checksum and inserts the checksum into the proper location within the packet before transmitting the packet on the memory. The network adapter also automatically separates headers and data during transfer of incoming packets from the adapter to the memory.

The network data further performs alignment of network headers by inserting pad bytes based on specific values found in the network link header. Col. 3, line 1-Col 4, line 50.

The network adapter is connected to the network through a front plane controller that provides transmission and reception of data packets to and from the network. For outbound transfers, the front plane controller unpacks the words from a DMA bus, looks at the first byte of the output stream, which contains a count of how many pad bytes were inserted in the packet and strips off the pad bytes. Col. 6, lines 35-46.

Scott teaches an improved system and method for transporting information over a communication channel. Scott uses a first frame 100 which includes a payload that includes user data PDU to which is prepended by a 4-octet ATM header that indicates that the frame is a low overhead cell frame. A trailer is also appended to the frame. Col. 8, lines 18-37. Scott also uses a second frame 150 which includes one of a plurality of 52-octet ATM cells to which is added a header, which indicates that the payload is framed cells, and a trailer. Col. 9, lines 10-27. The system includes a central transceiver which receives either frame 100 or 150 from a remote transceiver over a subscriber line. Figure 5C illustrates the steps performed at the central processor to implement a SAR (segmentation and reassembly) process. First, the payload is processed from frame 100 (block 231) and the number of octets of the user data PDU of the payload is counted (block 232). A user-to-user field and a common part indicator field are formed for the AAL5 frame (block 234). If the user-to user field and the common part indicator field are not included in the header or trailer, the default "0" is used. Pad characters are added

to make the AAL5 frame equal an integer number of 48 octet cells (block 236). The 32-bit cyclic redundancy check of the AAL5 frame is calculated (block 237) and the AAL5 frame is segmented into an integer number of 48 octet cells (block 238). Thereafter, the ATM header from the payload is extracted (block 239). A HEC is added to the 4 octet ATM header to form a 5 octet ATM header (block 241) which is prepended to the 48 octet cells (block 242). Col. 10, lines 16-58.

Parruck discloses that a multi-service segmentation and reassembly (MS-SAR) integrated circuit is disposed on a line card in a router or switch. The MS-SAR can operate in an ingress mode so that it receives packet and/or cell format data and forwards that data to either a packet-based or a cell-based switch fabric. The MS-SAR can also operate in an egress mode so that it receives data from either a packet-based or a cell-based switch fabric and outputs that data in packet and/or cell format. The MS-SAR has a data path through which many flows of different traffic types are processed simultaneously. Each flow is processed by functional blocks along the data path in accordance with one of several application types, the application type for a flow being predetermined by the host processor of the router or switch. Segmentation, reassembly and partitioning techniques are disclosed that reduce costs and facilitate high-speed operation. See at least the Abstract.

Applicants submit that the combination of Thompson, Scott and Parruck does not teach or suggest the combination of features clearly recited in the pending claims. Each of independent claims 1, 6 and 10, in part, recite an ingress module having an input

interface to receive a data packet including a plurality of cells, where a header cell of the data packet is one of the plurality of cells and the header cell of the plurality of cells includes a header and packet data information. Each of independent claims 1, 6 and 10 also recites that the header cell includes the header in its entirety for the data packet. Each of independent claims 1, 6 and 10 also recites, in part, inserting null bytes into the header cell of the data packet to form a modified header cell of the data packet if the counter determines that the header cell of the data packet does not satisfy the multiple of the predetermined number of bytes in order to align all of the other cells of the packet and removing the null bytes from the modified header cell of the data packet as a modified cell of the data packet exits the network device.

As acknowledged on page 4 of the Office Action, Thompson does not teach or suggest these features. Specifically, the Office Action acknowledged that Thompson does not teach or suggest inserting null bytes into **the header cell of the data packet to form a modified header cell** of the data packet if the counter determines that the header cell of the data packet does not satisfy the multiple of the predetermined number of bytes in order to align all of the other cells of the packet, as recited in claims 1-11. The Office Action also acknowledged that Thompson does not teach or suggest a counter.

Nevertheless, the Office Action cites Scott as curing these deficiency. Scott does not cure the deficiencies of Thompson, as noted above. There is no teaching or suggestion in Scott of inserting null bytes into the **header cell** of the data packet to form a modified header cell of the data packet if the counter determines that the header cell of

the data packet does not satisfy the multiple of the predetermined number of bytes in order to align all of the other cells of the packet, as recited in claims 1, 6 and 10. Figure 4 and Col. 10, lines 15-50 of Scott merely disclose that a frame includes a header portion and a payload portion. There is no teaching or suggestion in Scott of dividing the frame into a number of cells, with the header cell including header and data portions. As such, there is no teaching or suggestion in Scott of inserting null bytes into the **header cell** of the data packet to form a modified header cell of the data packet if the counter determines that the header cell of the data packet does not satisfy the multiple of the predetermined number of bytes in order to align all of the other cells of the packet, as recited in claims 1, 6 and 10.

In the "Response to Arguments" section, the Office Action alleged that Scott inherently discloses a data packet including a plurality of cells, where a header cell of the data packet is one of the plurality of cells and the header cell of the plurality of cells includes a header and packet data information. Scott discloses a traditional ATM cell where a data packet is divided in cells, each of the cells of the data packet including a header portion and a payload portion. There is no teaching or suggestion in Scott that the ATM cells includes only one header cell (for the entire packet) which includes the header in its entirety for the data packet, as recited in the pending claims. In the present invention, as recited in the claims, each data packet includes a plurality of cells with one header cell for all of the plurality of cells.

Parruck also does not cure any of the deficiencies of Scott and Thompson. Specifically, Parruck does not teach or suggest receiving a data packet including a plurality of cells, where a header cell of the data packet is one of the plurality of cells and the header cell of the plurality of cells includes a header and packet data information. The cited sections of Parruck do not teach or suggest that the header cell includes the header in its entirety for the data packet. In fact, Parruck also teaches the traditional ATM cell which is also disclosed in Scott. Col. 11, lines 15-19 of Parruck discloses that routing decisions are made based on the ATM **header in each cell**. Therefore, Parruck also does not teach or suggest the single header cell among a plurality of cells for a data packet, as recited in the pending claims.

Parruck also does not teach or suggest inserting null bytes into the header cell of the data packet to form a modified header cell of the data packet if the counter determines that the header cell of the data packet does not satisfy the multiple of the predetermined number of bytes in order to align all of the other cells of the packet and removing the null bytes from the modified header cell of the data packet as a modified cell of the data packet exits the network device. As noted above, the cited sections of Parruck merely disclose that a large ATM cell is divided into individual cells, each of the individual cells including a header and a payload portion. In fact, based on the teaching of Parruck, each data packet is divided into multiple header cells because each cell includes header and payload portions. There is no teaching or suggestion in Parruck of a header cell being one of a plurality of cells in a data packet, where the header cell includes a header and

packet data information and the header cell includes the header in its entirety for the data packet, as recited in the pending claims. Therefore, Applicants respectfully assert that the rejection under 35 U.S.C. §103(a) should be withdrawn because neither Thompson, Scott nor Parruck, whether taken singly or combined, teaches or suggests each feature of claims 1, 6 and 10. Each of claims 2-4, 7-8 and 11-12 depend on claims 1, 6 and 10 and should be allowed at least because of their dependence on claims 1, 6 and 10, in addition to the further limitations recited in claims 2-4, 7-8 and 11-12.

Claims 5, 9 and 13 were rejected under 35 U.S.C. 103(a) as being obvious over Thompson in view of Scott and further in view of Parruck and U.S. Patent No. 6,697,873 B1 to Yik. According to the Office Action, Thompson, Scott and Parruck teach all of the elements of claims 5, 9 and 13 except for teaching that the medium access control protocol module has a MAC address for transmitting the modified cell of the data packet and a layer two switching module configured to build a table for forwarding rules upon which the MAC address exists. Therefore, the Office Action combined the teachings of Yik with Thompson, Scott and Parruck to yield all of the elements of claims 5, 9 and 13. The rejection is traversed as being based on references that neither teach nor suggest the novel combination of features clearly recited in independent claims 1, 6 and 10, upon which claims 5, 9 and 13 depend.

Yik also does not cure the deficiencies of Thompson, Scott and/or Parruck, as outlined above. Yik teaches an apparatus and method for storing and searching computer node addresses in a computer network system. Each of claims 5, 9 and 13 depend on

claims 1, 6 and 10 respectively, and thus, incorporates all of the elements of the independent claims.

There is no teaching or suggestion in Yik of receiving a data packet including a plurality of cells, where a header cell of the data packet is one of the plurality of cells and the header cell of the plurality of cells includes a header and packet data information and the header cell includes the header in its entirety for the data packet, as recited in claims 1, 6 and 10 upon which claims 5, 9 and 13 depend. Yik also does not teach or suggest inserting null bytes into the header cell of the data packet to form a modified header cell of the data packet if the counter determines that the header cell of the data packet does not satisfy the multiple of the predetermined number of bytes in order to align all of the other cells of the packet and removing the null bytes from the modified header cell of the data packet as a modified cell of the data packet exits the network device, as recited in claims 1, 6 and 10. Therefore, Applicants respectfully assert that the rejection under 35 U.S.C. §103(a) should be withdrawn because neither Thompson, Scott, Parruck nor Yik, whether taken singly or combined, teaches or suggests each feature of claims 1, 6 and 10. Each of claims 5, 9 and 13 depend on claims 1, 6 and 10 and should be allowed at least because of their dependence on claims 1, 6 and 10, in addition to the further limitations recited in claims 5, 9 and 13.

Furthermore, Applicant respectfully notes that the Office Action has pieced together four references to teach the claimed invention. However, MPEP 2143.01 instructs that "[t]he mere fact that references can be combined or modified does not

render the resultant combination obvious unless the results would have been predictable to one of ordinary skill in the art. KSR International Co. v. Teleflex Inc., 82 USPQ2d 1385, 1296 (2007)." MPEP 2143.01 further instructs that "a statement that modifications of the prior art to meet the claimed invention would have been "well within the ordinary skill of the art at the time the claimed invention was made" because the references relied upon teach that all aspects of the claimed invention were individually known in the art is insufficient to establish a prima facie case of obviousness without some objective reason to combine the teachings of the references. Ex parte Levengood, 28 USPQ2d 1300 (Bd. Pat. App. & Inter. 1993)" Applicant respectfully submits that the cited references do not provide such an objective reason to combine the references. Applicant submits that the only objective reason to piece together the references of the Office Action is found in Applicant's own application. In addition, the references must be viewed without the benefit of impermissible hindsight vision afforded by the claimed invention.

In view of MPEP 2144.03, absent any teaching or suggestion in the prior art to adapt the teachings of Thompson to meet the claimed invention, and because the rejection lacks evidence of a teaching or suggestion that the features would have been obvious to one of ordinary skill, the rejections under 35 U.S.C. §103(a) are improper. Accordingly, Applicant respectfully submits that the rejections under 35 U.S.C. §103(a) should be withdrawn.

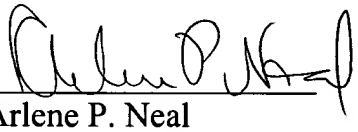
As noted previously, claims 1-13 recite subject matter which is neither disclosed nor suggested in the prior art references cited in the Office Action. It is therefore

respectfully requested that all of claims 1-13 be allowed, and this application passed to issue.

If for any reason the Examiner determines that the application is not now in condition for allowance, it is respectfully requested that the Examiner contact, by telephone, the applicant's undersigned attorney at the indicated telephone number to arrange for an interview to expedite the disposition of this application.

In the event this paper is not being timely filed, the applicant respectfully petitions for an appropriate extension of time. Any fees for such an extension together with any additional fees may be charged to Counsel's Deposit Account 50-2222.

Respectfully submitted,



Arlene P. Neal
Registration No. 43,828

Customer No. 32294
SQUIRE, SANDERS & DEMPSEY LLP
14TH Floor
8000 Towers Crescent Drive
Tysons Corner, Virginia 22182-2700
Telephone: 703-720-7800
Fax: 703-720-7802

APN:skl